



Multi-constrained routing algorithm: a networking evaluation

van Dan Nguyen, Thomas Begin, Isabelle Guérin-Lassous

► To cite this version:

van Dan Nguyen, Thomas Begin, Isabelle Guérin-Lassous. Multi-constrained routing algorithm: a networking evaluation. IEEE HSNCE 2013, Jul 2013, Kyoto, Japan. pp.5. hal-00864322

HAL Id: hal-00864322

<https://hal.science/hal-00864322>

Submitted on 23 Sep 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multi-constrained routing algorithm: a networking evaluation

Van Dan Nguyen*, Thomas Begin* and Isabelle Guérin Lassous*

*Université Lyon 1 / LIP (UMR INRIA, ENS Lyon, CNRS, UCBL), 46 Allée d'Italie, 69007 Lyon, France

Email: thomas.begin@ens-lyon.fr, isabelle.guerin-lassous@ens-lyon.fr

Abstract—IP networks may face issues to support the offered workload due to the increasing number of Internet users, the steady influx of new Internet applications, which require stringent QoS, and applications needing big data transmission. QoS routing can be viewed as an attractive approach to tackle this issue. However, most of the QoS routing solutions are not evaluated in a realistic framework. In this paper we propose a networking evaluation of multi-constrained routing to assess the potential benefit of QoS routing protocol. To do this, we converted a multi-constrained routing algorithm into a protocol, and implemented it in the simulator NS2. Our results indicate that if the monitoring tool of a network can not sustain frequent link-state announcements, the benefits coming from implementing a QoS routing are quite low. On the other hand, if the network is equipped with an adequate measurement tool, then QoS routing can be worth implementing, and the routing based only the available bandwidth at each link arises as the best option (no need to consider the end-to-end delay constraint, nor the loss rate constraint).

Index Terms—QoS routing; Quality of Service; Traffic classes; Performance evaluation.

I. INTRODUCTION

Back in the 1970s, Internet was thought to support network communications without any performance guarantees. The original idea was simply to provide a best-effort delivery service to a single class of traffic. However, Internet has since undergone numerous changes. The number of users has been growing worldwide, the number of resource-intensive applications, *e.g.*, big data transmission, is constantly going up, and recent applications, such as Voice over IP, peer-to-peer video exchange or real-time multimedia applications, with strict Quality of Service (QoS) requirements, have emerged. These applications must also coexist with seminal Internet applications, such as mail or remote access.

Several solutions have been proposed to cope with these profound changes while preserving the current networking infrastructure. These include techniques like IntServ, DiffServ, MPLS and Admission Control. Some of them, *e.g.*, MPLS paths, are frequently deployed in backbone networks, but they do not provide a comprehensive QoS solution. Another approach, simple and widely used by operators, is to overprovision the network so that the links capacity goes far beyond the expected workload needs [6]. However, this approach may face limitations in the long run. First, it might be too expensive in the future to systematically overprovision a network as

the network workload increases. Second, speeding up the performance of all networking communications, when only a specific subset of them needs to be improved, can be seen as an inefficient use of resource for operators.

Providing QoS to flows in both an efficient and scalable way still remains as a challenging issue for network operators. Several operators and researchers agree that the core of next-generation networks should be designed with the idea of providing a native support to applications with tight constraints on bandwidth, loss rate or end-to-end delay [10]. In fact, future networks should, on one hand, ensure accurate and tunable guarantees to each individual flow requesting QoS, and on the other hand, from the operators point of view, be simple to deploy, easy to manage and allow a high level of network resource utilization. Existing QoS solutions, based on best-effort routes, such as DiffServ and IntServ [15], [16], can hardly meet these objectives. QoS routing can then be viewed as an attractive approach to tackle this issue since it can carefully compute routes with specific requirements. Some authors view QoS routing as the missing piece in a comprehensive QoS architecture for the Internet [9], [14], [5]. Of course, other new solutions such as Software-Defined Networking (SDN) could also be of a great help into this goal.

Many QoS routing algorithms have been proposed in the literature [3], [7]. These algorithms are usually derived as solutions to the Multi-Constrained Optimal Path problem (MCOP). In their framework, authors consider a cost or, more generally, a vector of costs for each link of the network. Such vectors of costs aim at reflecting the current state of the link through several metrics. Typical metrics include the available bandwidth (*i.e.*, residual capacity), the delay experienced by a packet over this link (including queueing delay) and the loss rate (*i.e.*, the probability that an incoming packet is dropped due to the buffer overflow). However, most of these algorithms are confined to a theoretical framework so that their evaluation only pertain to the algorithm evaluation with regards to its computational complexity and memory requirement [11]. Others use numerical simulations to assess the effectiveness of their algorithm in a more realistic network context [4], [13]. However, these simulations may fail to capture some significant aspects of computer network. First, the initial values for the link costs and the QoS requirements of flows are often chosen randomly. Second, when a new flow attempts to access a link, the new cost of this link is simply

This work has been partially supported by the project *Semantic Networking* within the common laboratory INRIA / Alcatel-Lucent Bell-Labs.

computed as being its previous value to which is subtracted the actual value requested by the new incoming flow. Though summing up costs could be a roughly fair approximation when costs pertain to the available bandwidth, such an assumption is obviously no longer true for parameters like delay or loss rate. Third, models used to represent the network traffic can be seen as unrealistic. In some cases [4], network traffic is simply described by the vector of its QoS requests, and not by an actual flow of packets. Finally, the stale link-state problem, which impairs QoS effectiveness due to out-of-date collected announcements, is barely considered, except in a couple of articles, reviewed in [8]. Overall, QoS routing algorithms are rarely evaluated as QoS routing protocols, and this shortage makes it difficult to fairly assess the potential benefit of QoS routing solutions. We conclude this state of the art with the work of Shaikh et al. They provided a practical evaluation of their routing solution [12]. However, as their QoS routing was only designed to handle bandwidth requests, their results apply to a single-constrained routing scheme.

This paper aims at providing first results to bridge the gap between the theoretical behavior of QoS routing algorithms, as provided by their authors, and a more practical focus on their performance and limitations if they were to be deployed in real IP networks. To do this, we converted a well-known multi-constrained routing algorithm, namely PIRA [4] [7], into a multi-constrained routing protocol, and implemented it in the simulator NS2. This enables us to investigate how much a QoS routing protocol can help in meeting the needs of applications with firm transmission requirements. Our contributions are twofold. On the one hand, our results indicate that QoS routing can be effective even if we only consider the flow requirements in terms of data rate. On the other hand, if the frequency at which the link-state announcements are generated and broadcasted is too low, then QoS routing becomes of little help, whatever the considered costs are. The remainder of this paper is organized as follows. In Section II, we describe our investigation framework. Section III is devoted to the performance evaluation results. Section IV concludes this paper.

II. INVESTIGATION FRAMEWORK

In this section, we describe the framework we use to assess the potential benefits of using a QoS routing protocol. It includes the selected multi-constrained routing algorithm and the different building blocks we implemented in order to make the algorithm closer to a real routing protocol.

A. Traffic classes

A simple and commonly used approach to classify the Internet traffic is to consider two types or classes of Internet traffic: *elastic traffic* and *inelastic traffic* [1].

Elastic traffic requires no specific QoS, and therefore only expects - at least to a certain extent - the routing algorithm to find a feasible path between sources and destinations. Flows into this class obtain unspecified variable bit rate, delivery time and loss rate, which depend on the current traffic workload.

In our framework, we assume that network operators simply provide a best-effort service to flows from this class.

By contrast, inelastic traffic is vulnerable to delay and packet loss. It is made up of flows that require specific levels of QoS. Typical examples include audio and video flows since those latter have firm requirements in terms of end-to-end delays and losses. In our experimental framework, whenever a new inelastic flow is created, the network computes an “ad hoc” path based on a multi-constrained routing algorithm.

B. Multi-constrained routing algorithm

As said before, the search for paths satisfying QoS requirements is performed thanks to PIRA [4]. PIRA is a link-state algorithm based on an “All Hops k -shortest Paths” algorithm that basically proceeds in two phases: a first phase that iteratively computes all hops k -shortest paths using a single cost defined as a linear combination of the different costs contained in the QoS requests; and a second phase that selects a feasible path from the set of paths computed in the first phase. The greater k , the more likely PIRA will find a satisfying path (assuming its existence). The authors of PIRA show that, with k equal to 2, their heuristic almost always returns a feasible path (if any exist) [4]. In our experiments, we choose k equal to 3 so that PIRA does not miss a feasible path during its search (provided it gets up-to-date knowledge on the link-states). If no adequate route is possible, then the incoming flow is simply refused.

C. Update policy

PIRA being a link-state algorithm, the state of each link (*viz.* its cost) must periodically be advertised to each node of the network. Of course, the cost of each link is dynamic (it varies with time) since it depends on the amount of traffic traversing it. We now describe how each node of the network gets (more or less recent) information regarding the states of the networks links. First, each node continuously monitors the state of its adjacent links, and computes the mean observed values for the available bandwidth, the packet delay, and the packet loss rate. Second, link-state announcements, which comprise these vectors of costs, are periodically broadcasted across the network. Obviously, the time that elapses between two link-state announcements deeply affects the paths computed by PIRA. We consider different values for this update period. For the ease of readability, we express the time period between two consecutive updates in terms of the number of inelastic flows that attempted to enter in the network. We denote this number of flows by i . Thus the two extreme scales are (i) updates are sent just prior to the introduction of a new flow into the network, *i.e.*, $i = 1$, and (ii) no update at all will occur during the experiment, *i.e.*, $i = \infty$.

D. Routing policy

To route inelastic traffic, we consider all routing schemes resulting from the possible combinations of the three described metrics used to assess the actual cost of a link, namely the Bandwidth, the Delay and the Loss rate. For the sake of

Cost / Metric	Corresponding constraint	Routing schemes
Bandwidth	$\min_{k \in s(P)} B_k > b$	<i>BDL</i> , <i>BD</i> , <i>BL</i> and <i>B</i>
Loss rate	$1 - (\prod_{k \in s(P)} (1 - L_k)) < l$	<i>BDL</i> , <i>BL</i> , <i>DL</i> and <i>L</i>
Delay	$\sum_{k \in s(P)} D_k < d$	<i>BDL</i> , <i>BD</i> , <i>DL</i> and <i>D</i>

$s(P)$ denotes the set of links that constitute the path P .

TABLE I
LIST OF THE CONSTRAINTS FOR QoS ROUTING SCHEMES.

readability, we denote each scheme by the capitals of the metrics that are actually involved in the search for the path. For instance, the *BDL* scheme refers to the case where PIRA takes into account the whole link costs measurements. On the other hand, the *DL* scheme relates to the case where only the delays and loss rates reported values matter for the QoS routing. Hence, seven schemes have been studied: *BDL*, *BD*, *BL*, *DL*, *B*, *D*, *L*. Finally, we also include the case where PIRA simply disregards the update reports, and therefore, route inelastic flows just as elastic traffic, *i.e.*, through the shortest path in terms of number of hops towards the destination. We denote this scheme by *H*. Of course, we expect that the more knowledge the nodes get, the better the QoS routing will perform.

E. Formal statement of the problem

Let f be an incoming inelastic flow whose source and destination are nodes A and B , respectively. f has a data rate of b and has strict constraints both on its packet loss rate and on its end-to-end delay. We denote by l and d their maximum allowable values, respectively. If the performance of f happen to go beyond any of these two values, the QoS of f is then declared to be unmet.

The QoS routing works as follows. Based on its latest received link-states announcements (that reflect the state of each network link j in terms of the available bandwidth, B_j , the delay experienced by packets, D_j , and the rate of loss, L_j), the QoS routing looks for a path P between nodes A and B so that P satisfies the QoS requirements of f . We report the set of possible constraints in Table I. Obviously, depending on the QoS routing scheme, the set of (operating) constraints differ (see Table I).

If an appropriate path is found for f , then we rely on simulation experiments to determine if the actual performance of f , once injected in the network, meet its QoS requirements, namely if its loss rate and its end-to-end delay are less than l and d , respectively. The outcome of this test step can be negative due to variations in the network workload, and to the staleness of measurement data.

III. PERFORMANCE EVALUATION

A. Simulation-driven approach

To assess the potential benefit of QoS routing, we conduct a simulation-driven study. We equip the network simulator NS2 with a multi-constrained routing protocol, derived from PIRA algorithm, as explained in Section II. This implementation

Inelastic applications	Source data rate	Performance requirements	
	Bandwidth	End-to-end delay	Loss rate
Audio (A)	80 Kbps	150 ms	4%
Video 1 (V1)	0.5 Mbps	150 ms	4%
Video 2 (V2)	1 Mbps	150 ms	4%
Video 3 (V3)	3.5 Mbps	150 ms	4%

TABLE II
INELASTIC FLOWS WITH THEIR QoS REQUIREMENTS.

Position	1	2	3	4	5	6	7	8	9	10
Type	A	A	V2	V2	V2	A	A	V2	A	V3
Position	11	12	13	14	15	16	17	18	19	20
Type	A	A	V3	A	V2	A	V3	V1	V3	A

TABLE III
FIRST SEQUENCE USED TO INTRODUCE INELASTIC FLOWS IN THE NETWORK.

enables us to investigate the actual performance of QoS routing under various settings.

B. Network and traffic condition

Our network roughly follows the topology that was initially proposed by Chen [2], and subsequently used by several other authors. This topology, depicted in Figure 1, aimed at representing the backbone of a north-american network operator. We set the links attributes as follows. The links exhibit bandwidth capacities of 10 Mbps or 50 Mbps (represented in Figure 1 by plain and bold lines, respectively), a buffer size of 50 packets, and a propagation delay randomly set to 2, 5 or 10 ms.

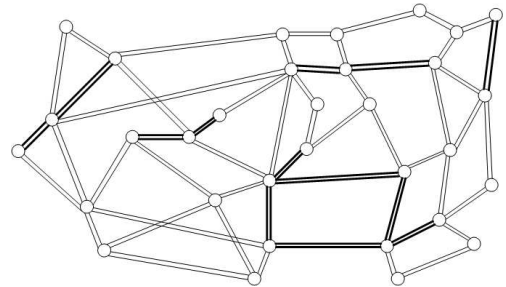


Fig. 1. The network topology.

We use Poisson sources to represent the class of elastic traffic. Over each (unidirectional) link, a Poisson source, delivering 1000 bytes packets, generates workload at a constant rate. The rates of sources are uniformly distributed between 50, 60 and 70% of the link capacity (*i.e.*, bandwidth). With such rates, packets may be exposed to queueing delays in the link buffers and also to buffer overflows. Hence, we set up the elastic traffic so that, the network is not over-provisioned and so that is not straightforward to provide the requested QoS to inelastic flows (*e.g.*, the end-to-end delay between two random nodes along the shortest path in terms of hops can often exceed 150 ms due to long queueing delays).

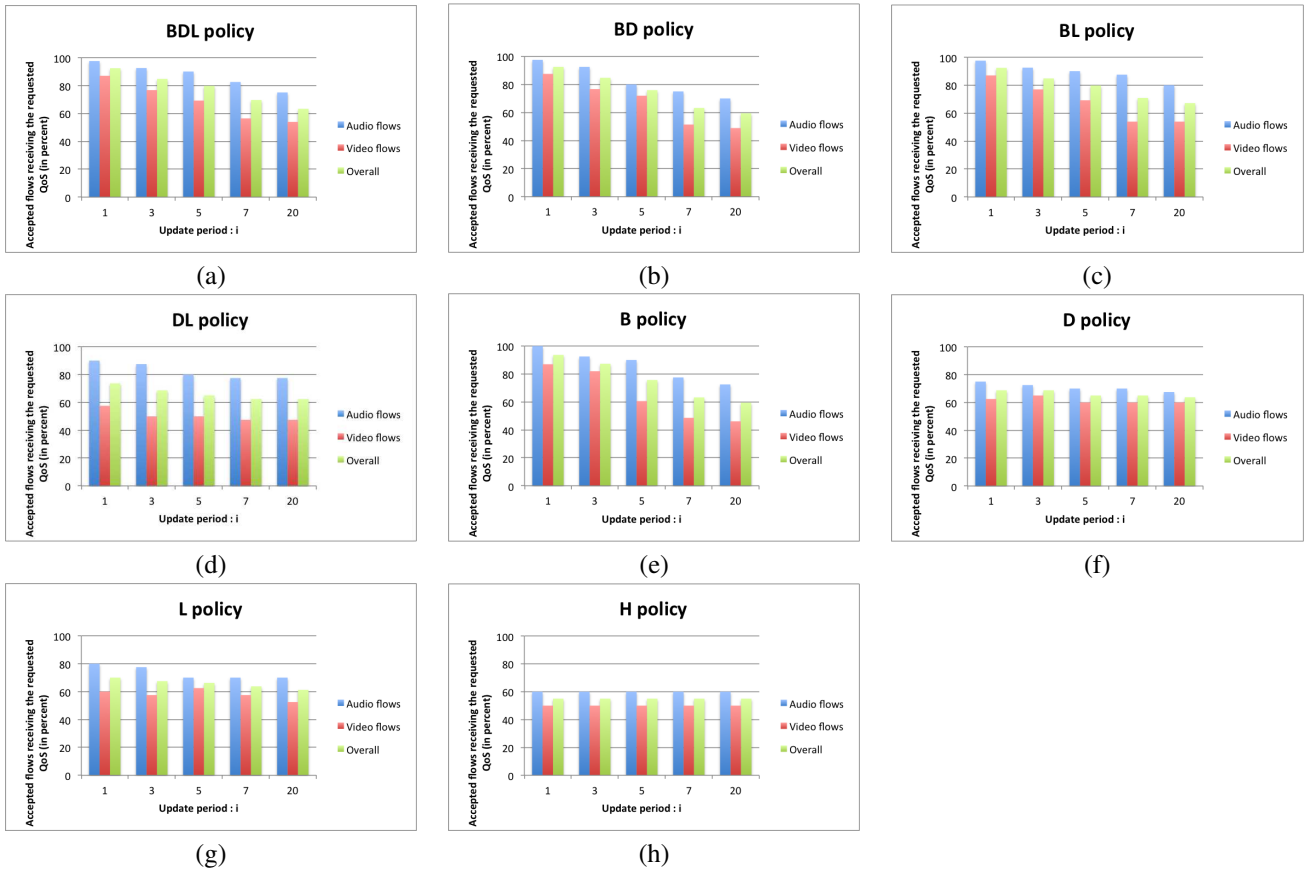


Fig. 2. Performance of QoS routing schemes as a function of length of the update period.

Inelastic traffic is modeled by CBR (constant bitrate) sources that send data to nodes that are several hops away. In our simulations, we consider 20 inelastic flows whose source and destination are randomly chosen among the network nodes. These CBR flows have various sending rates and QoS requirements (see Table II). The length of packets for the flows corresponding to audio and video applications are of 200 and 940 bytes respectively. These values correspond to QoS levels as required by typical codecs for real-time voice and video [1]. We consider four possible sequences for the introduction of inelastic traffic within the network (already loaded with elastic traffic). The details of the first sequence is reported in Table III.

C. Numerical results

We have performed simulations of the latter scenario for all the possible schemes of QoS routing and for different levels of staleness of the link-state information (i varies between 1 and 20, since for each sequence at most 20 inelastic flows are injected into the network). Remind that i determines the period that elapses between two link-state announcements. If this update period is too long, then there is a risk for inelastic flows to be processed along inadequate routes, that may severely hamper their performance. This is known as a “false positive”. On the other hand, “false negatives” occur when the network refuses to route a flow, though, there was a feasible route. In our experiments, false negatives rarely occur. At the end of

each simulation, we compute the proportions of inelastic flows that receive their requested QoS during all the simulation time.

Figure 2 shows the corresponding results with the mean values obtained through our 4 independent sequences of introduction of inelastic flows. Each subfigure relates to a specific QoS routing scheme. It reports the proportion of (inelastic) flows that, once accepted in the network, receive the requested QoS as a function of the update period, i . We have represented this proportion when considering only audio flows (smaller data rate), only video flows (larger data rate) and both. For example, Figure 2(a) refers to the *BDL* case where the QoS routing uses all the information provided by the update reports, namely the bandwidth, delay and loss rate link-state announcements.

First, we observe from Figure 2(a) that audio flows, which have smaller sending rate, tends to outperform video flows. This observation, which also applies for all other scenarios, is not surprising since a lower data rate for a flow tend to less affect the network and its links.

Second, as expected, as long as the length of update period, *i.e.* i , increases, the performance of the QoS routing decreases. For instance, in case of *BDL*, the percentage of flows that meet QoS falls from 92% when every new incoming flow is preceded by a update on the link measurements to a level around 63% when no update at all occur. This trend also applies for the other schemes. However, this trend is more

pronounced for cases such as *BDL*, *BD*, *BL*, *B* than for the others, namely *DL*, *D* and *L*. Of course, in case of the scheme *H*, the proportions of accepted flows receiving their QoS is independent of *i* (see Figure 2(h)). In fact, in case of a pure shortest path scheme, only slightly more than half of inelastic flows receive their requested QoS.

IV. DISCUSSION AND CONCLUSION

We now discuss our results and their potential implications when planning the use of a multi-constrained routing. We elaborate our conclusions based on the observations of the previous section as well as other experiments that were run with different settings with respect to the network topology, the levels of workload and the buffer size of the communication links. Although the quantitative results may vary, the qualitative conclusions remain roughly the same.

First, our experiments show that in order to satisfy almost all accepted flows, say more than 95% of them, QoS routing requires to rely on more than a single metric. Second, as expected, the frequency at which the nodes send their link-state announcements is a critical factor for QoS routing. However, our results indicate that QoS schemes dealing with two or more metrics tend to be quite sensitive to the update period value, while simpler QoS policies dealing with only one metric seem to exhibit a similar behavior for a larger range of time period between link-state announcements. Third, interestingly, the so-called *B* QoS routing scheme, which only relies on the link-state announcements dealing with the available bandwidth on each link, does not comply with these two latter rules. Indeed, this quite simple *B* routing policy performs just as well as the best of the other policies though its performance also tend to decrease fast when the update period increases. Therefore, given the configuration of today's IP networks, QoS routing on the available bandwidth of each link emerges as the best tradeoff among all QoS routing schemes and it can provide excellent results provided the update period is small enough. There is no need to consider the end-to-end delay constraint, nor the loss rate constraint since their inclusion does not seem to improve the efficiency of QoS routing. More generally, the good behavior of the *B* routing policy seems to support the idea that bandwidth is the critical resource in an IP network, and that the other performance metrics such as delay and loss rate will be met as long as the former is.

To sum up, if the monitoring tool of a network can not sustain frequent link-state announcements (*i.e.*, the period spacing two consecutive measurement reports is too long), the benefit coming from implementing a QoS routing is quite low. On the other hand, if the network is equipped with an adequate measurement tool, then QoS routing can be worth implementing, and the routing based only on the available bandwidth at each link arises as the best option (no need to consider the end-to-end delay constraint, nor the loss rate constraint).

REFERENCES

- [1] Braun, T., Diaz, M., Enríquez-Gabeiras, J., Staub, T.: End-to-End Quality of Service over Heterogeneous Networks. Springer-Verlag, ISBN 978-3-540-79119-5 (2008).
- [2] Chen, S., Nahrstedt, K.: On finding multi-constrained paths. Proceedings of IEEE ICC, 2, 874-878 (1998).
- [3] Chen, S., Nahrstedt, K.: An overview of quality-of-service routing for the next generation high-speed networks: problems and solutions. IEEE Network, Special Issue on Transmission and Distribution of Digital Video (1998).
- [4] Cheng, G., Ansari, N., Papavassiliou, S.: Adaptive QoS provisioning by pricing incentive QoS routing for next generation networks. Computer Communications, Elsevier, Issue 31, pp 2308-2318 (2008).
- [5] Crawley, E., Nair, Raj., Rajagopalan, B., Sandick, H.: A framework for QoS-based routing in the internet. Internet RFC 2386 (1998).
- [6] Fraleigh, C., Tobagi, F., Diot, C.: Provisioning IP Backbone Networks to Support Latency Sensitive Traffic. Proceedings of INFOCOM 2003, 375-385 (2003).
- [7] Garroppo, R.G., Giordano, S., Tavanti L.: A survey on multi-constrained optimal path computation: Exact and approximate algorithms. Computer Network 54, issue 17, 3081-3107 (2010).
- [8] Guérin, R. A., Orda, A.: QoS routing in networks with inaccurate information: theory and algorithms. IEEE Transactions on Networking, 7(3) (1999).
- [9] Masip-Bruin, X., Yannuzzi, M., Domingo-Pascual, J., Fonte, A., Curado, M., Monteiro, E., Kuipers, F., Van Mieghem, P., Avallone, S., Ventre, G., Aranda-Gutiérrez, P., Hollick, M., Steinmetz, R., Iannone, L., Salamatian, K.: Research challenges in QoS routing. Computer Communications 29, 563-581 (2006).
- [10] Meddeb, A.: Internet QoS: Pieces of the Puzzle. IEEE Communication Magazine, 48(1), 86-94 (2010).
- [11] Van Mieghem, P., Kuipers, F.: Concepts of Exact QoS Routing Algorithms. IEEE/ACM Transactions on Networking, 12(5), 851-864 (2004).
- [12] Shaikh, A., Rexford, J., Shin, K. G.: Load-Sensitive Routing of Long-Lived IP Flows. Proceedings of SIGCOMM, 215-226, Cambridge, Massachusetts, USA (1999).
- [13] Spittler, S. L., Lee, D. C.: Integration of Explicit Effective-Bandwidth-Based QoS Routing with Best-Effort Routing. IEEE/ACM Transactions on Networking, 16(4), 957-969 (2008).
- [14] Wang, Z., Crowcroft, J.: Quality-of-Service Routing for Supporting Multimedia Applications. In IEEE Journal on selected areas in communications, 14(7), 1288-1297 (1996).
- [15] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss W.: An Architecture for Differentiated Services. IETF, RFC 2475, December 1998.
- [16] Braden, R., Clark, D., Shenker, S.: Integrated Services in the Internet Architecture: an Overview. IETF, RFC 1633, June 1994.